A LOW LEVEL SYNTACTIC CHECKER AND AUTO CORRECTOR FOR INDIAN LANUAGES

Rushikesh K. Joshi*, Hemant N. Patil, Sagar A. Tamhane

Department of Computer Science and Engineering

Indian Institute of Technology Bombay

Mumbai - 400076

Abstract

Individual letters in Indian languages can be compositions of multiple ligatures. We identify a set of typing errors associated with the composition rules. These errors are low level syntactic errors. This paper discusses these errors and the design and implementation techniques for a tool which detects and eliminates low level syntactic errors in Indian languages. The language chosen for this paper is Marathi which uses Devanagari alphabet.

1 Introduction

Spelling mistakes in English are word level typing errors and are detected with the help of a dictionary. An English word is composed of letters which are themselves atomic ligatures such as 'A' and 'G'. In contrast to this, in an Indian Language, a letter may be composed of multiple atomic ligatures such as **प**, **अ**, **हलंत** character ् **रफार** character र्‍ and **मात्रा** characters. Hence while typing in an Indian Language typing errors at letter level may also occur. We call these errors as low level syntactic errors. In subsequent sections we present an analysis of various types of low level syntactic errors with examples and provide guidelines for building tools for detection and autocorrection. A tool has been built by the authors at IIT Bombay which resulted in removal of numerous low level typing errors on a typed text Marathi. The tool was built for ISCII [1] and Akruti_Priya_Expanded [2] documents. We also discuss some aspects of design and implementation of this tool and its performance.

*Contact Author, Email: rkj@cse.iitb.ac.in

2 Types of Low Level Syntactic Errors

Below, a classification of low level syntactic errors in a document written in an Indian Language is provided with a few examples. Reasons of these errors may be attributed to violations of the constraints of multiplicity, ordering, and composition of ligatures, especially of the matra characters.

2.1 Multiplicity

Typically, ligatures such as matra, halant and anuswar are used only once in a single letter composition. While typing, it may be possible that they get typed twice and the result is still not visible in the text in an Indian Language font. Such errors can be classified as invisible errors, while some remain visible in the display.

Invisible Errors

For example, in the case of an extra anuswar in letter अं, the source in ITRANS is given by aM, whereas, aMM also produces अं. The Devanagari characters seen in the previous sentence were actually produced by two different encodings. While the result of this error is not visible in the display, this could be a cause of concern for an application such as a pattern matching algorithm, an index or a converter.

Invisible errors may also occur in the case of multiple ukar and matra (ए and ऐ) characters. For example, in पू , an extra ukar is present but it is invisible, or overlapped.

Visible errors

While some ligatures overlap in display and the errors remain undetected by mere human inspection of the display, some errors may be detectable by display inspection. This is possible due to positioning adjustments done on ligature placements.

For example, an extra kana in Marathi, such as in letter पा , is visibly detectable if it occurs more than once since the additional ligature is right shifted and either displayed as a kana, or taken as a separate character. An example of the later case is encoding paaa in ITRANS, which is displayed as पाअ.

2.2 Ordering

In some cases, more than one vowels can occur in a single letter. For example, an anuswar and a kana in Marathi as in letter पां
may be typed in Akruti_Priya_Expanded in two ways, as hebe (where h stands for प् , e for kana and b for anuswar), or as heeb. While the first choice results in an unaesthetic placement of anuswar on top of the first

vertical line, the second choice gives the desired result as in पां . Another example is that of पो which may be typed as hebe or heeb. With a smaller font size, the differences may get overlooked, but with an enlarged font size, they are easily detected. Similarly, ordering is important in character र्पा . It may be typed as here or as heer.

2.3 Mutual Exclusion

In Indian Languages, the composition rules are also specified for mutually exclusive vowels. For examples, an ukar (i.e. उ) and a kana (i.e. आ) cannot occur in one character. Similarly, a velanti and an ukar or a kana cannot occur in one letter. Whereas, it is possible for a kana (i.e. आ) and a matra ए to occur together in one letter.

2.4 Association

Certain consonants do not accept rafar (i.e. र्) or specific conjuncts. For example, a combination of ळ and र् is invalid. Similarly, it is not possible to combine certain consonants such as ज्ञ with other consonants, or ह with प.

3 Low Level Syntactic Checker and Autocorrector

There are two approaches to implementing a low level syntax checker, either by specifying all valid combinations, or by specifying all invalid compositions. Genericity may be exploited. For example, some of the errors listed in earlier section are generically applicable over the entire alphabet.

We built a low level syntactic checker and autocorrector based on the ideas discussed in this paper. The tool with a set of rules was applied to तुकारामांची गाथा in project Tukaram [3], which contains around 4600 verses in Marathi containing over 2,00,000 words. and over 300 errors were detected and automatically corrected by this tool, proving the utility of the low level automatic syntactic correction.

4 Conclusion

A classification of low level syntactic errors in Indian Languages was presented. Various kinds of errors were analyzed with examples. Guidelines for building a tool for detection and correction of these errors were also discussed. An application on a pretyped text proved the utility of the ideas discussed.

5 References

1. Bureau of Indian Standards, Indian Script Code for Information Interchange - ISCII, Indian Standard IS 13194:1991, 1991.
2. Akruti Software, Akruti_Priya_Expanded Font, *http://www.akruti.com*
3. Centre For Indian Language Technology Solutions, Searchable and Browsable Collection of Tukaram's Abhangas, http://www.cfilt.iitb.ac.in/tukaram/index.html, Indian Institute of Technology Bombay, 2001.